****

日本語アブスト近年、

**

非暗号的メディア処理による擬似ブラインド電子透かし*

岡田満雄 †, 岡部寿男 ‡, 上原哲太郎 ‡

Pseudo-Blind Watermarking Based on Non-Cryptographical Media Processing*

Mitsuo OKADA†, Yasuo OKABE‡ and Tetsutaro UEHARA‡

A concept of pseudo-blind fingerprinting is proposed in this paper. The pseudo-blind fingerprinting functions effectively for a practical and secure content trading. A prototype of a web-based image trading system is implemented. We focused on a problem that most of existing secure content trading schemes have not reached at practical level despite the demand of secure trading is increasing. One of the primal reasons is its difficulty to satisfy both feasibility and security simultaneously which are often inconsistent with each other. Our goal is to provide a concept that leads to a scheme to achieve practical level of feasibility and security. Digital fingerprinting is one of the content protection techniques in which the client's security is guaranteed only under the premise that a content provider was perfectly trustworthy. Such premise makes a scheme unpractical. Blind fingerprinting schemes have been proposed in which cryptography is utilized for the sake of client's security. However, hybridizing cryptography and fingerprint embedding involves many restrictions that make these schemes unpractical because of insufficient robustness of watermark and heavy cryptographical computation cost. The proposed pseudo-blind fingerprinting resolves the difficulties by applying a combination of a media process blinding method and fingerprinting to protect the client's information and content instead of using cryptography. Existing watermark algorithms can be well hybridized without restrictions because the proposed blinding method and watermark embedding are both media processing manipulations. We have implemented a prototype in which one of the media processing methods described as image decomposition has been implemented. This prototype assures both a purchaser and a provider of fair trading that is effective in a market where the purchasers have to deal with small or not so reliable content providers. User-friendly operation that requires no special tools, skill nor knowledge to handle this system is also our priority concern for the sake of a serviceable application. The evaluation results concerning robustness of fingerprint and perceptual condition of an image prove validity of the scheme.

1. Introduction

Recently, all kinds of digital content can be purchased through the Internet such as music, image, and book accompanying a rapid development of IT infrastructure. At the same time, enormous amount of digital content might have been pirated because the protection has not well considered in the existing content trading systems even though demand for protecting intellectual property of digital content is increasing due to the severe crime augmentation. In fact, Multimedia Intelligence (US) reports that demand of DRM (Digital Right Management) products including information hiding technologies would surpass 500 million US dollars by 2012 [4]. One of the primal reasons that inexistent of the secure models is its difficulty to satisfy both feasibility and security simultaneously which are often inconsistent with each other. Our goal is to provide a concept that leads to the scheme to achieve practical level of feasibility and security.

The current major existing content trading models are that 100% trustworthy content providers (provider) distribute digital content to huge number of purchasers (client). In this scheme, a client pays for content to a provider, but copyright of content and the client's privacy are insecure. For example, client's privacy such as name, credit card number, and purchasing records provided at the payment process is revealed to the provider. Therefore, the privacy of the client could be easily violated if the provider were malicious, whereas unprotected content delivered to the client could be easily pirated if the client were malicious.

* **20090101

† Graduate School of Informatics, Kyoto University; Yoshida-Honmachi, Sakyoku, Kyoto, JAPAN

‡ ACCMS, Kyoto University

Key Words: Privacy protection, Blind fingerprinting, Content trading application.

Digital fingerprinting [19] using watermarking techniques is one of the effective approaches to protect digital content. A provider embeds a client ID into original content before deliver it to the client. If pirate content had been found, a malicious client were identified by extracting the embedded ID from the fingerprinted content in which ID is embedded. However, this scheme has privacy and non-repudiation problems. The former one is that client's information could be revealed to a malicious provider. The later one is that if the provider wasn't trustworthy, a malicious party who redistribute content were unable to be identified since exactly the same fingerprinted content was possessed by both the client and the provider.

The above models are not effective in a market where a number of subdominant or not so reliable small providers distribute content. Generally speaking, small providers are somehow lacking in trustworthiness as compared with such providers as the one owned by well known enterprises. Therefore, both a provider and a client must be able to observe dishonesty in trading.

A TTP (a trusted third party) model in which a trusted third party is interposed for fingerprint embedding and client verification may be one of the solutions for privacy and non-repudiation problems. However, this is not effective because privacy information is exposed to TTP, and TTP's managing cost of privacy information will be expensive.

Blind fingerprinting models [16,17,5–8,2] resolve the privacy and non-repudiation issues. In these methods, client's privacy is protected at high level of security by applying a combination of cryptography and fingerprinting. However, they are impractical because of following reasons.

- Fingerprint tends to be fragile because fingerprint embedding is restricted by cryptography.
- Cryptographical computation cost is expensive.
- A client is required to prepare and well manage a cryptography key.

As described the above, existing secure content trading schemes have not reached at practical level because of feasibility and security issues.

As a countermeasure to the above defect, we propose pseudo-blind fingerprinting which effectively functions to practical and secure content trading [12,11,10]. The pseudo-blind fingerprinting uses a non-crypt blinding method in order to enhance practicality. For example, information separation by media processing is one of the effective methods. Media processing methods indicate high-pass filtering, color inversion, gamma correction, and so on in which most of existing fingerprinting algorithms can be well hybridized. It results in fingerprint to be robust because fingerprint embedding is also media processing manipulation.

The idea of pseudo-blind methods can be applied to any multimedia digital content such as image, movie, music, and so on. In order to show the feasibility, we have implemented a prototype of an image trading system in which information separation by media process described as image decomposition has been applied. Both security and feasibility are achieved in order to provide fair trading for the sake of both a provider and a client. TTP who handles fingerprint embedding and client verification is interposed in the prototype. Privacy information contains client's identities and purchasing history such as who purchased what. In order to protect the privacy, the scheme is designed to allow TTP to know client's identities such as name, an anonymous ID, and email address, but not purchasing history such as what kinds of content has been purchased. On the other hand, a provider is allowed to know purchased content, but not client's identities. Hence, a client is able to purchase content without exposing privacy information to both a provider and TTP. In order to show the total performance, we have evaluated the prototype of a content trading system.

The conventional blind fingerprinting methods with their defects are described in section 2.. Our contributions are described in following order, the proposed pseudo blind fingerprinting in section 3., prototype and detail of technical elements in section 4., evaluated result in section 5., and conclusion in section 6..

2. Conventional Fingerprinting Models

Conventional fingerprinting models are described in this section. In the naive content trading methods, a client pays to a provider and obtains purchased content. However, unprotected content could be easily pirated by a malicious client, whereas the client's privacy managed by a provider could be easily violated by a malicious provider.

2.1 Basic Model of Digital Fingerprinting

Digital fingerprinting [19] based on a digital watermarking technique is one of the approaches to protect digital content against piracy. In a basic model of content trading scheme, a client pays for content by sending a client ID, *cID* and payment information such as credit card number or digital cash. As soon as the payment is approved, the provider embeds *cID* into an original content I to generate a fingerprinted content as $I' = I \oplus cID$ and then sends it to the client as illustrated in Fig. 1. Note that $\oplus$ indicates fingerprint embedding operation. When a pirated content had been found on the Internet or some website, a provider extracts *cID* to identify a malicious client.

This model is effective only if the provider were perfectly trustworthy. Otherwise, inappropriate trading could be easily carried out by a malicious provider since both a provider and a client possess the same fingerprinted content. Even if a pirated content had been found, a malicious party either the provider or the client couldn't be identified. Another problem is that a malicious provider could violate client's privacy such as name, credit card number as well as purchasing history.

2.2 Conventional Blind Fingerprinting

Blind fingerprinting schemes [16,17,5,15] have been proposed for the sake of secure trading for a provider and a client. A combination of cryptography and fingerprinting is applied to protect client's privacy and provider's content.

A basic idea of the conventional blind methods is described below. In order to compare the conventional blind

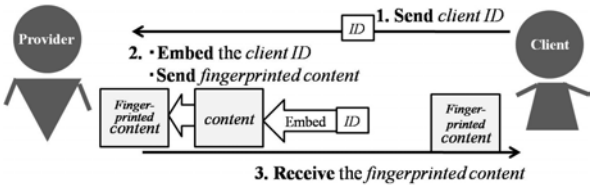


Fig. 1 Basic Model of Fingerprinting

methods with our pseudo-blind method, we apply blind methods to a content trading scheme based on a TTP interposal framework. TTP is interposed to deal with fingerprint embedding and privacy management of a client. An illustration is shown in Fig. 2. The index numbers in the figure and following description are synchronized.

(1) A client prepares a pair of public and secret keys, (pk, sk) for homomorphic cryptography and then obtains an anonymous ID, *aID* from TTP. pk is used for encryption and sk is used for decryption. The client sends *aID* and pk to a provider to purchase content. The provider verifies *aID* in cooperation with TTP by sending *aID* to TTP. TTP checks validity of *aID*. If verified, TTP returns the verification result to the provider.

(2) Assume that *aID* is verified, the provider encrypts original content I using the pk as $E_{pk}(I)$ and then sends $E_{pk}(I)$ and *aID* to TTP. Note that $E(\cdot), D(\cdot)$ are homomorphic encryption and decryption respectively.

(3) TTP encrypts *aID* as E_{pk}(*aID*) and then embeds it into the encrypted content without decryption as $E_{pk}(I') = E_{pk}(I) \oplus E_{pk}$(*aID*). $\oplus$ can be additive homomorphic calculation for embedding and I' represents fingerprinted content. The encrypted fingerprinted content $E_{pk}(I')$ is sent to a client.

(4) The client decrypts $E_{pk}(I')$ by sk to obtain the fingerprinted content as $I' = D_{sk}(E_{pk}(I'))$.

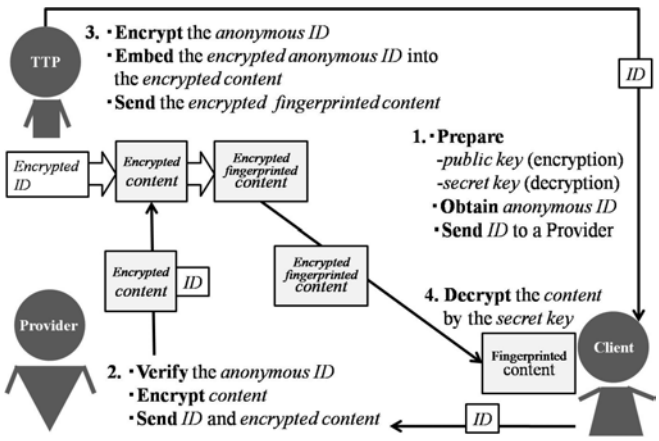


Fig. 2 Basic Model of Blind Fingerprinting

This scheme resolves privacy and non-repudiation problems. Purchasing history is protected by allowing TTP to handle the encrypted content. The identity information is protected by allowing a provider to handle anonymous ID. A non-repudiation problem that is identifying a true illegal party has been resolved since the fingerprinted content can be obtained only by the client who possesses the secret key sk. Some conventional blind methods are introduced below.

2.2.1 Based on Bit-Commitment

A bit-commitment technique is used in [16,15] for watermark extraction. In these methods, computation and communication cost increases in proportion to the size of an image that makes a scheme inefficient. According to [7], 1GB of ciphertext is needed to extract watermark from 1MB of an image. It is indicated in [3] that robustness is not sufficient because XOR operation used for embedding can be easily removed by compression.

2.2.2 Based on Homomorphism Encryption

We show some blind methods based on homomorphism encryption. Okamoto-Uchiyama encryption is used in [7]. Message is embedded by modifying pixels using a combination of QIM (Quantization Index Modulation) and Okamoto-Uchiyama encryption.

According to [18], it is indicated that simple QIM lacks in robustness of watermark, and it involves heavy degeneration of an image. To resolve these defects, DC-QIM (Distortion Compensate QIM) is applied in [18] for robust embedding. However, they are still impractical due to robustness and computation cost caused from cryptography.

A combination of patchwork watermarking and Paillier encryption is used in [2] and El Gamal in [8] respectively. In these methods, index of modified pixels which is used for embedding are encrypted as an extraction key. Watermark can be extracted by using the extraction key without exposing the index information so that a provider has no clue as to where watermark is embedded. However, the size of the ciphertext, an extraction key and computation cost increases in proportion to the size of an image.

According to [8], if 1024 bits of Paillier encryption had been applied to ($z = 256 \times 256$) pixels of an image, the size of chippertext would be ($1024 \times z$) bits and encryption time would be approximately ($3.3 \times z$) seconds where single bit encryption takes 3.3 seconds. Watermark tends to be fragile since watermark algorithms as well as its capability would be much limited due to the restriction caused from cryptography.

For the case of 1024 bits of El Gamal encryption, the size of ciphertext would be ($1024 \times 2 \times z$) bits. Note that two chipertext are generated in El Gamal. For example, the chippertext becomes 50MB in order to embed 1 bit. Estimated embedding time is approximately ($0.1 \times z$) second where a single bit encryption takes 0.1 second. For example, it takes approximately 175 minutes for embedding and 1.5 minutes for extracting.

Implementation environment of [2] and [8] is summarized in Table 1. Cryptographical computation is implemented in Java, and watermark manipulation is implemented in C. Refer [8] for more detail.

As a summary of conventional blind methods, the problems of the non-repudiation and the privacy issues have been achieved, but robustness of watermark and cryptographical computation cost should be concerned toward practical applications.

Table 1 Implementation Environment

Detail	Specification
CPU	Xeon 2.3GHz
OS	Redhat 9.0, Linux 2.4.20
Memory	1GB
Encryption Algorithms	1024-bit El Gamal, 1024-bit Paillier
Programming Languages	J2SDK 1.4.2, gcc 3.3.3

3. Proposed Pseudo-Blind Method

Redundancy is indispensible element for digital fingerprint and watermark embedding since the message is embedded in the redundant area of the content. Generally, multimedia content contains large amount of redundant information which can be often omitted at compression process. Even though redundant information is omitted, the content is not perceptually degenerated.

In contrast with multimedia content, the redundancy in cryptography is considered as a conflict. Therefore, the conflict causes cryptography and fingerprinting to be ineffectively hybridized. For example, little media process manipulation in multimedia content may appear just as noise which hardly influences the whole meaning of the content. On the other hand, even single bit of modification or error in a cryptosystem affects the whole meaning of the context unless a complement technique such as error correction code had been applied. We propose a method that resolves the defect by altering cryptography with a media processing approach.

3.1 Definition of Pseudo-Blind Method

Our proposed pseudo-blind methods [12,11,10] which are an alternative method to conventional blind methods are the solution for the inefficiency. The comparison of blind methods and pseudo-blind methods is described below with an illustration in Fig. 3.

Blind methods use cryptography to protect information in which security relies on cryptography. TTP embeds fingerprint into encrypted content. Therefore, TTP cannot guess what the content is. On the other hand, we defined pseudo-blind methods that use information separation by media processing instead of using cryptography. In this scheme, separated data is delivered to a client by different routes. Fingerprint is embedded in one of the separated data by TTP. Therefore, only partial information is exposed to TTP at embedding process. Security of which relies on difficulty in reversing media process and profiling the original content from the partial incomplete data. Hence, complete blindness is guaranteed in blind methods, but not in pseudo-blind methods. Instead of the security issue, pseudo-blind methods are superior in feasibility.

Our proposed pseudo-blind methods lead to achieve practical level of feasibility and sufficient level of security since fingerprinting and blinding techniques are well hybridized. In addition, most of existing watermarking methods can be applied with almost no restriction. This is effective because they are designed to be robust against media processing attacks. A concept of pseudo-blind method is applicable to any multimedia content, but in order to show the feasibility, we have implemented an image trading scheme based on the concept.

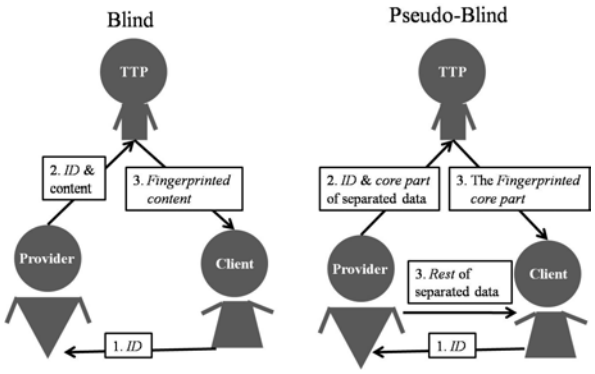


Fig. 3 Comparison of Blind and Pseudo-Blind Method

3.2 Pseudo-Blind Based Image Trading Scheme

In this section, we describe an image trading scheme based on the pseudo-blind method. Blind methods provide complete blindness in which an entire image is encrypted. On the other hand, pseudo-blind methods blind up only necessary parts of an image by image manipulation as media processing such as high-pass filtering, color inversion, gamma correction, and so on. In other words, pseudo-blind methods allow exposing partial information but blinds up information which needs to be protected such as recorded subjects, image detail, or color information in an image. For example, human face and license number in the image is blinded in [9] by clipping human face or masking an entire image by noise. Instead of security issue, the feasibility such as computation cost and robustness of fingerprint in pseudo-blind methods is more efficient than blind methods as summarized in Table. 2.

Feasible and secure content trading is effective for a market where a number of small providers distribute digital images to clients such as a CGM (Consumer Generated Media) market. Clients may have counteraction against small providers since they, generally speaking, are somehow lacking in trustworthiness compared to well known providers. Therefore, both a client and a provider should be able to observe dishonesty each other.

In the scheme, client's privacy is protected by separating the privacy information into identity and purchasing history which are sent to a provider and TTP respectively. Privacy is protected by allowing TTP to know who, but not what kind of image has been purchased, whereas a provider is allowed to know what, but not who.

Content is protected by digital fingerprinting. The information separation requires two distribution channels for content delivery. Following conditions should be satisfied in order to achieve privacy secure and content secure at the same time.

- Purchasing history should be protected against TTP.

- A client's identities should be protected against a provider.
- Content should be protected against a client.

One of the effective approaches is our proposed image decomposition method based on a pseudo-blind fingerprinting framework as illustrated in Fig. 4. Both of blind and pseudo-blind methods are able to protect the client's privacy against the other parties, but they vary in blindness level and feasibility.

The procedure of the scheme is described below. In prior to trading, a client obtains an anonymous ID, *aID* from TTP. The client requests an image by using *aID*. Assume that the client is anonymously verified by the provider in cooperation with TTP.

(1) The provider decomposes an image into a complement piece (I_c) and an endorse piece (I_e). I_c should have no resale value while I_e is required to be unrecognizable by human eyes in order to avoid TTP from profiling the purchased image. I_c is sent directly to a client and I_e with *aID* is sent to the client via TTP.
(2) TTP embeds *aID* into I_e and then sends it to the client.
(3) The client integrates the two decomposed images to obtain a complete fingerprinted image.

If a suspicious image had been found by a provider, the image is forwarded to TTP for *aID* extraction as $(I, aID) = EXT(I')$ where *EXT*$(\cdot)$ is a fingerprint extraction process. If *aID* had been extracted, TTP discloses the client information.

The scheme provides following achievements.

- A provider has no access to client information besides an anonymous ID and purchasing records.
- Even though TTP knows identities of the client and the anonymous ID, TTP has difficulty to profile the purchased content by guessing original condition purchased content from I_e at embedding process.
- If pirated content were found, a malicious party could be certainly distinguished as the client since the only client could obtain the complete fingerprinted image. There is no way for a provider to obtain the fingerprinted image.
- A client is able to purchase an image without being exposed privacy information.

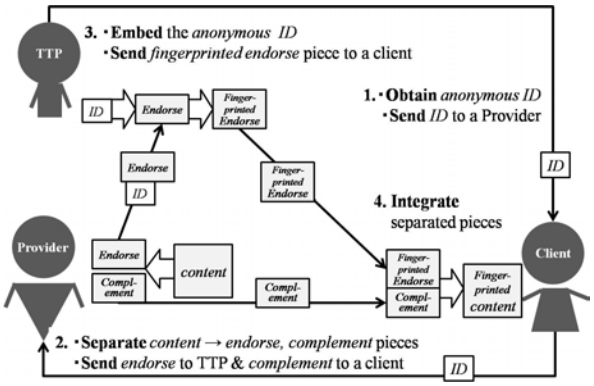


Fig. 4 Pseudo-Blind Fingerprinting

3.3 Requirements of Pseudo-Blind Fingerprinting

Pseudo-blind methods are mainly composed of image decomposition, fingerprint embedding, and image integration as summarized below.

3.3.1 Requirement for Decomposition

Decomposition is required to satisfy two inconsistent conditions simultaneously that is generating unrecognizable I_e in which fingerprint should be able to be robustly embedded. However, the problem is that an unrecognizable image preserves less image information such as details of recorded objects. Therefore, embedding capacity would be very limited. In other words, fingerprint embedded in complete noise image does not survive through an integration process because I_e and I are completely different. It means fingerprint embedded in the features of content would be damaged by integration process.

For example, fingerprint would be robustly embedded in an image labeled as "Low Blind Level" in Fig. 12, but an image is very identical since each block contains enough information to recognize the image. Yet, fingerprint is not well embedded in the image labeled as "High Blind Level" which is much unrecognized.

3.3.2 Requirement for Embedding

Primal considerations of fingerprint embedding are robustness against image integration and watermark removable attacks. Fingerprint should not get damaged by integrating two image together. The embedded fingerprint should be survived through watermark removable attacks by media process manipulation.

3.3.3 Requirement for Integration

A primal consideration of the integration process is minimum side effect of embedded fingerprint. Integration should not damage the embedded fingerprint by the integration process.

In addition to the side effect, we consider the usability of this tool toward practical applications. We assume that this tool is used by ordinary people. Therefore, this integration operation should requires no a special tool, skill, nor knowledge to the client. It has been accomplished by a wrapping over integration method in which a transparent image is wrapped over another image to recompose a complete fingerprinted image. The detail is described in section 4.1.4.

4. Prototype of the Image Trading System

In this section, a prototype of privacy-secure image trading system based on the pseudo-blind fingerprinting is presented. Collusion attacks among the entities are not concerned in this paper. We assume that a provider is small or not so reliable. Therefore, a client doesn't want to reveal her/his privacy such as name to the provider. TTP is a broker between the provider and the client who embeds fingerprint to protect content and manages client's identities. Even though TTP is trustworthy, the client doesn't want to reveal purchasing history to TTP.

Table 2 Comparison of Blind and Pseudo-Blind Methods

Descriptions	Blind Method	Pseudo-Blind Method
Framework	Based on cryptography	Based on media processing
Applicable watermarking	Additive embedding method	Any existing methods
Fingerprinting capacity	Low	High
Robustness of fingerprint	Fragile	Robust
Security measurement	Rely on cryptographic robustness	Rely on reversing media process such as mosaic, image decomposition, and so on.
Processing cost	Heavy	Light
Defect	Heavy computation cost, difficulty in key management	Incomplete blindness against TTP

4.1 Procedures in the Prototype

This scheme is mainly composed of verification, trading, image integration procedures. In the verification procedure, client is verified by the provider in cooperation with TTP. In the trading procedure, the client obtains an image from the provider in cooperation with TTP. In the image integration procedure, a client integrates two images into a complete fingerprinted image.

An image of 512×512, 256 gray levels, 11-bit of message length, 15-bit of the codeword length are used. For example, 11-bit of decimal numbers (2001) which can be selected from $1-2047$ as an anonymous ID *aID* is expanded to 15-bit of binary code. ω is fingerprint message that includes error correction code.

4.1.1 Verification Procedure

First, a client obtains *aID* through the registration page provided by TTP. An anonymous client verification method has not been specified in this paper, but an existing federated identity framework such as Shibboleth [1] may be suitable.

Second, the client login to a provider's webpage using *aID* where an image can be purchased. At this point, TTP possesses the client name and the anonymous ID, whereas the provider only possesses the anonymous ID and a purchasing history. Therefore, TTP has difficulty to profile what kind of image has been purchased, whereas the provider has no clue as to who the client is.

4.1.2 Purchasing Procedure

Usability of this tool is described and shown in Fig. 5

(1) Assume that the client had been successfully verified as an anonymous user.
(2) A client moves to the page where an image can be purchased by clicking one of the thumbnails.
(3) Trading procedure which contains image decomposition and fingerprint embedding processes are executed and then two decomposed images are generated.
(4) Finally, the client receives the two images, endorse piece and complement piece from TTP and the provider respectively. The images are integrated to be a complete fingerprinted image as described later on.

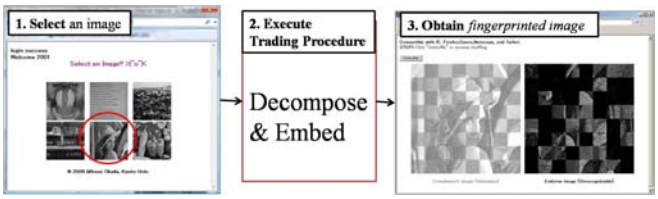


Fig. 5 Purchasing Procedure

4.1.3 Trading Procedure

The detail of the trading procedure is described below. The index numbers in Fig. 6 and following instruction is synchronized. We assume that a client had been clicked a thumbnail to select an image.

(1) A provider receives the HTTP post from a client.
(2) As soon as a provider receives the inquiry, image is decomposed into I_c, a complement piece and I_e, an endorse piece as *DCMP* as $(I_c, I_e) = DCMP(I)$. I_c is allowed to be accessed by a client, whereas I_e is allowed to be accessed by TTP. I_e is number of small $bc \times bc$ pixels of blocked images, $(I_{e_1}, \ldots, I_{e_{bn}}, bn = (Col/bc * Raw/bc))$ as shown in the figure. In this implementation, $bn = 64 = (512/64 * 512/64)$ of small blocked images are generated from a 512×512 pixels of an image.
(3) The provider returns HTML content as a response to the inquiry of the client. The HTML content contains links to I_c and I_e. Former one is a single link to I_c while the latter one contains multiple links to small blocked images $(I_{e_1}, \ldots, I_{e_{64}})$.
(4) Then HTML content traces the links to obtain two images. I_c is obtained directly from the client. I_e is obtained from the provider via TTP. When TTP receives the HTTP post, TTP gets all the blocked images $(I_{e_1}, \ldots, I_{e_{64}})$ from the provider and then embeds fingerprint into the selected blocks. The detail of an embedding process is described later. All the fingerprinted blocks are forward to the client.
(5) A client obtains two images from the provider according to the links. The integration procedure is described later on.

4.1.4 Integration Procedure

Final step is generating a complete fingerprinted image from the two decomposed images by a client as $I' = INTG(I_c, I'_e)$ where *INTG* is an image integration function. Integration in this prototype is composed of two simple steps as shown in Fig. 7. This is designed to be user-friendly [13] in which no special skill, tools, nor knowledge is required.

First, reverse shuffling the endorse piece according to the shuffling key possessed by the provider. Next, resize the frame of a browser to overwrap an image one another [1].

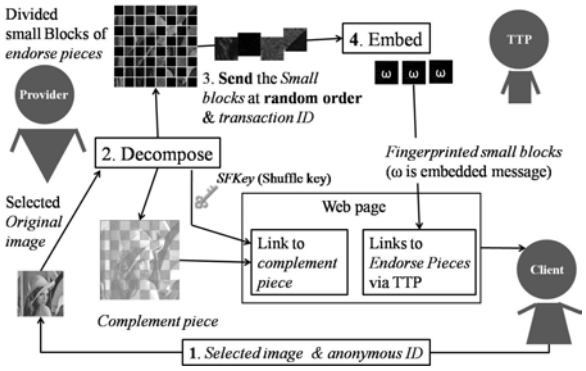


Fig. 6 Trading Procedure

4.2 Media Process Blinding Method

Technical details of image decomposition, *DCMP* is described below with some output images in Fig. 8. Note that brightness the high-pass filtered part has been modified in order to depict the condition. The decomposed images should be satisfied following requirements. I_c is required to be valueless so that a client has no incentive to resale or redistribute it. I_e is required to be unrecognized, but fingerprint should be robustly embedded so that TTP is hardly guess what kind of image has been traded.

We have applied three decomposition elements; frequency decomposition (*FQD*$(\cdot)$), area division (*ARD*$(\cdot)$), and invisible masking (*IVM*$(\cdot)$) as an image decomposition function (*DCMP*).

4.2.1 Consideration of Fingerprint Embedding

FreQuency Decomposition, *FQD*$(\cdot)$ is function that extracts complicated area in an image such as an edge of recorded subjects by high-pass filtering as $I_H = FQD(I)$. The counterpart is generated as I_L =*SUB*$(I$,*FQD*$(I))$. *SUB* is subtraction operation of pixel by pixel of two input images. For example, it subtracts pixels of *FQD*(I) from those of I.

Fingerprint is effectively embedded in the complicated area of an image since small manipulation is hardly noticed by human eyes. For example, brightness modification of single dot in hair part is almost impossible to recognize, but the modification in the skin is easily recognized.

FQD$(\cdot)$ should be applied to the original image in order to extract a pure high-pass component. In other words, if other decomposition elements had been applied before *FQD*, noise or block border affects extracting the high-pass component. Ordinary DCT is used in *FQD* with 8×8 small blocks. The detail of a high pass filtered image is not discussed in this paper.

4.2.2 Consideration of Blindness

ARea Division, *ARD*$(\cdot)$ breaks up the entire image detail since the original condition of the image may be easily profiled from the high-pass filtered image.

Block-check image division is effective one that divides an image into two groups of square-block-check images as I_{b1} =*ARD*(I), and the counterpart as I_{b2} =*SUB*(I, I_{b1}). The block-check image contains clipped parts and blank parts which appease black box as shown in 8 labeled as (Block check image). The images can be integrated by Area division inverse function, *BI*$(\cdot)$ that integrates active blocks, the other parts of the blank parts.

Random shuffling function, *RS*$(\cdot)$ is applied to shuffle block order in the block checked image. A shuffle key is generated when shuffling which is used to reverse shuffling. It contains index of randomly shuffled block order.

In order to secure the blindness, the image might divide in the landscape or portrait depends of the image. For example, seashore scene image would be more effective to divide in the landscape. In this implementation, we show square blocks division. Square blocks contains equal amount of frequency component horizontally and vertically.

In this prototype, *ARD*$(\cdot)$ divides an image into small $zn \times zn$ blocks to generate number of blocks as $Z_1, \ldots, Z_\eta$ where $\eta = (Col/zn \times Row/zn)$ is the number of blocks in the image. For example, $(\eta = 64 = (512/64 \times 512/64), zn = 64)$ small blocks are generated from a $512 \times 512 = Col \times Row$ image.

4.2.3 Consideration of Valueless Property

InVisible Masking (*IVM*$(\cdot)$) makes an image I_c valueless so that a client has no incentive to redistribute I_c before receiving I_e. In addition to the valueless property, *IVM*$(\cdot)$ could blind up information such as characters or rough image of human faces remained in I_e.

One of the effective methods is a pseudo noise image generation. A noise image is adaptively generated according to an original image. For example, when the image contains less high-frequency, heavy noise is applied.

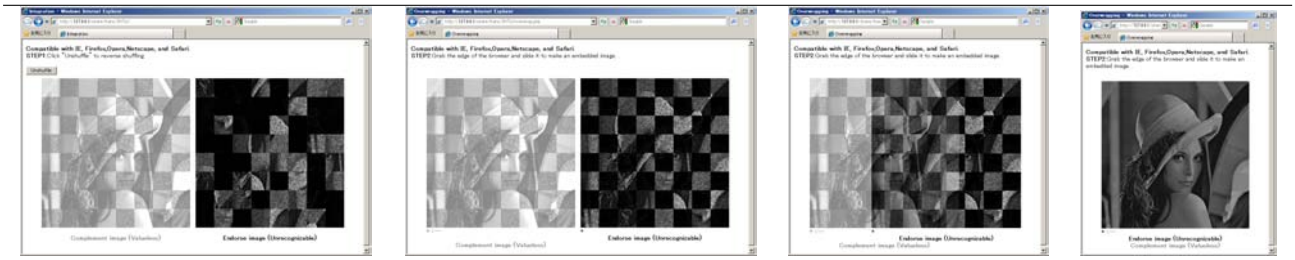
Two pseudo-noise images are generated as I_{n1} =*IVM*(I) according to the input image, and the counterpart as I_{n2} =*SUB*(I, I_{n1})

These noise images are integrated by summing up brightness values of two images each other. Therefore, the noise generation should be controlled in order to avoid overflow at integration process.

A block noise image I_{n1} is generated as described below.

(1) Determine block size for block noise and find the minimum brightness value from the blocks. In this prototype, the size is $16 = bn \times bn, bn = 4$.
(2) Generate a pseudo random number to generate block noised image and assigns it to the pseudo noise image. The noise is generated from range between 0 to the minimum value in order to avoid overflow when adds the two images together. For example, if the

[1] Wrapping over can be tested in `www.net.ist.i.kyoto-u.ac.jp/watermark/INTG`



1. Transparent I_c and Shuffled I'_e 2. Reverse shuffling of I'_e 3. Over wrapping 4. I'

Fig. 7 Integration *INTG* Procedure

minimum brightness is 150, the pseudo noise should be 0 to 150.

(3) Iterate this process for number of blocks. In this prototype, the block number is $16384 = 512/4 \times 512/4, bn = 4$ for a 512×512 image.

4.2.4 Image Decomposition Procedure (*DCMP*)

Image DeCoMposition, *DCMP* is composed of the above elements that outputs an endorse piece of I_c and a complement piece of I_e as described below.

(1) Generating a high-pass filtered image, $I_H = FQD(I)$ and the counterpart as $I_L = SUB(I, I_H)$. Note that $SUB(\cdot)$ is substation function that subtract brightness value I_H from I. Even though I_H is hardly recognized, the detail of an entire figure is somewhat visible. Furthermore, main component of an image remains in I_L which is worth resale or product level. Therefore, we apply other decomposition elements.

(2) Generate a block-check image as $I_{Hb} = ARD(I_H)$ to break up the detail of an entire figure in I_H.

(3) Deteriorate I_L to make a valueless image, I_{Ln} as $I_{Ln} = IVM(I_L)$.

(4) Input I_{Ln} into $ARD(\cdot)$ to generate block checked noised low-pass image I_{Lnb} which is used to make I_e.

(5) Generating an endorse piece I_e. Two block checked images are merged as $(I_{Lnb} I_{Hb} = BI(I_{Lnb}, I_{Hb})$ and then shuffles the blocks by block shuffling function, $RS(\cdot)$ as $I_e = RS(I_{Lnb} + I_{Hb})$. A shuffle key that contains index of original block order is managed by the provider.

(6) A complement piece, I_c is generated from the counterpart of $(I_{Hb} + I_{Lnb})$ as $I_c = SUB(I, (I_{Hb} + I_{Lnb}))$ where $SUB(\cdot)$ represents substation of brightness of two images.

Finally, I_c and I_e is sent to a client and TTP respectively.

4.3 Fingerprint Embedding

The pseudo-blind fingerprinting scheme for fingerprint embedding has been so designed that most well-developed existing embedding techniques can be applied with little restriction unlike the conventional blind methods. Therefore, embedding can be altered to other existing watermarking algorithms.

In this prototype, we apply a watermark algorithm that embeds message in frequency domain named as Coefficients Comparison Embedding (*CCemb*). I_e contains two types of blocks, high pass filtered block Z_h and noise Z_n. *CCemb*, effective for high-pass filtered blocks, is applied to Z_h. No embedding is applied in Z_n in this prototype, but some other embedding can be applied. For example, patchwork watermarking had been applied in [10].

The same message ω is embedded redundantly and independently into all blocks, Z_h. The blocks are classified either Z_h (high-pass) or Z_n (noise) by finding average brightness ψ in every blocks. If ψ were larger than a threshold υ, the block would be classified as Z_h in which *CCemb* is applied.

CCemb embeds fingerprint in the selected blocks which contains a high complex part in the image as shown in Fig. 9.

(1) Block Z_k is divided into small $M \times M$ blocks $Y_\ell, \ell = 1, \ldots, (zn/M \times zn/M)$. In this implementation, $zn = 64, M = 8$ are used. zn is the block size of the block checked image by *ARD*.

(2) High-complex part is selected by a block analysis function $BA(\cdot)$ which finds the blocks Y_ℓ containing complicated area in Z_k. Detail of $BA(\cdot)$ is described later on.

(3) γ pairs of coefficients are modified to represent 1 bit. The group of pairs (A, B) in a block Y_ℓ are chosen as $A = a_1, \ldots, a_\gamma, B = b_1, \ldots, b_\gamma$ from the selected blocks Y. It is required to satisfy large $dst = |A| - |B|$ for selecting coefficient. dst is distance between a_i, b_i that enhances detection accuracy. Small dst easily affects watermark to be removed. The coefficients $a_i, b_i, i = 1, \ldots, k$ are randomly selected from a range of low to middle and middle to high frequency domains respectively in the blocks Y. k is the index number of selected blocks.

(4) a_i and b_i are modified to embed watermark as

$$\omega = \begin{cases} 0 & A < B, \\ 1 & A \geq B. \end{cases} \quad (1)$$

The two values are switched and parameter δ is added to adjust dst for robustness to each pairs as

$$\omega = \begin{cases} 0 & a_i - \delta \ll b_i + \delta, \\ 1 & a_i + \delta \gg b_i - \delta, \end{cases}$$

to satisfy the condition (1).

(5) Continue the above process according to γ and length of ω. For the case of this implementation, totally ω (bits)$\times\gamma$ pairs of coefficients from each selected

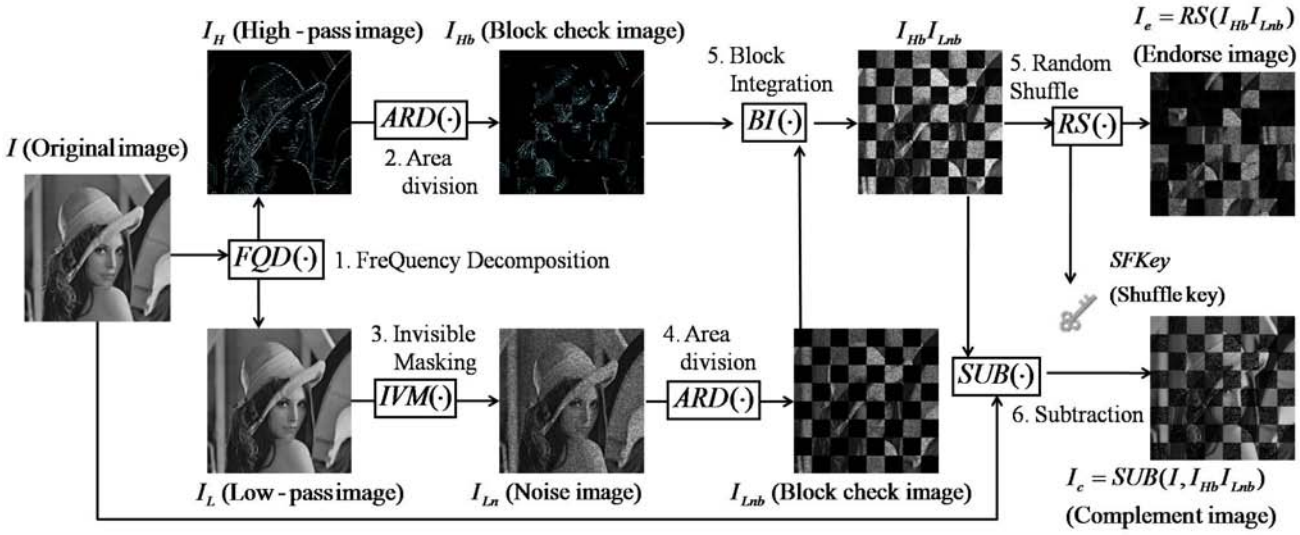


Fig. 8 Image Decomposition *DCMP* Procedure

blocks Z_k are modified. In this paper, $15(\omega)+30(\gamma)$ pairs of coefficient are modified in every blocks Z.

$BA(\cdot)$ is a function in embedding to enhance detection accuracy. It finds small blocks Y_ℓ in Z_k containing large standard deviation $\sigma \gg \tau$. It indicates the blocks Y contains high-complexity area in Z_k where small modification is hardly recognized by human eyes. Note that τ is a threshold to determine complexity. Increment of τ indicates higher robustness at heavy image degeneration at embedding and vice versa. In other words, Y_ℓ containing large σ provides better detection accuracy, whereas the one possessing small σ involves false detection.

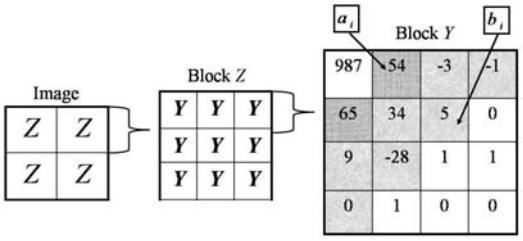


Fig. 9 Coefficients Selection in $M \times M$ blocks Y

5. Evaluation

Perceptual and robustness evaluations are shown in this section. The former one shows perceptual condition of decomposed images and fingerprinted image. The latter one shows robustness of the fingerprint. Parameters used in this implementation and its environment are summarized in Table 3 and 4.

5.1 Perceptual Evaluation

Perceptual evaluation for decomposition using various types of images is shown in Fig. 10 (From top to bottom; *Baboon*, *Book*, *Granada*, *Kyufun*, and *Peppers*). Note that *Baboon* and *Peppers* are provided by USC SIPI database. The other images are prepared by the authors.

A complement piece of I_c directly sent to the client has no resale or product value while an endorse piece of I_e sent to the TTP for embedding is hardly recognized by human eyes.

Table 3 Environment

Detail	Specification
CPU	Intel Xeon E5345 2.33GHz
Memory	4GB RAM
OS	Fedora 10
DCMP, *EMB*	Matlab2009a
Web interface *INTG*	HTML, PHP

Table 4 Parameters

Parameters	Values
Original Image	512×512 pixels, 256 Gray levels
aID	11-bit in decimal
Fingerprint ω	15-bit in binary
Redundancy γ	30
Robustness δ	20

I'_e is shown in Fig. 11 in which high strength-level of embedding has been applied to show distinct embedding effects. Therefore, I'_e is heavily degenerated.

5.2 Evaluation Using Different Parameters

Parameters can be adjusted for security levels. The difficulty in restoring shuffled blocks of I_e can be increased by making the block size smaller. Unrecognizability of I_e is increased by adjusting noise levels and high-pass levels as shown in Fig. 12. It shows the result that as I_e approaches higher unrecognizability, I_c approaches to better quality. Watermark strength would be decreased as unrecognizability of I_e increases.

5.3 Robustness Evaluation of Watermark using Stirmark

Stirmark [14] is a benchmark tool that generates various manipulated images of the fingerprinted image. Pa-

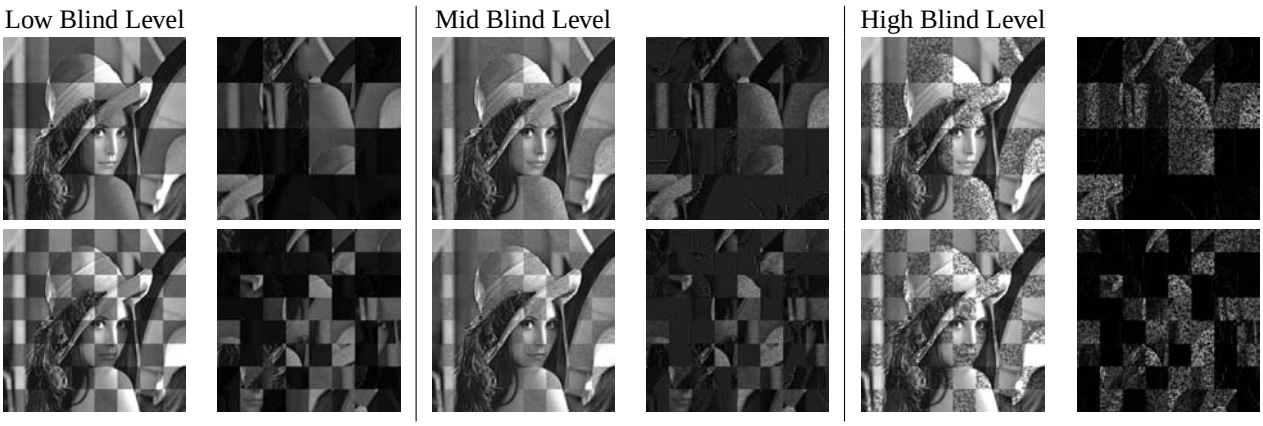


Fig. 12 Images using Various Parameters

Table 5 Parameters of Stirmark and Results

Attacks	Description	Number of Attacks	Levels	Succeed
AFFINE	Affine Transform	8	$1,2,\ldots,8$	None
CONV	Gaussian Filtering	2	$1,2$	All
CROP [%]	Cropping	3	$25,50,75$	None
JPEG[%]	JPEG compression	7	$20,30,\ldots,80$	$30,\ldots,80$
MEDIAN	Median cut	4	$3,5,7,9$	3
NOISE[%]	Add noise	8	$10,20,\ldots,80$	None
RESC [%]	Rescale	6	$50,75,90,125,150,200$	All
RML [lines]	Remove lines	9	$10,20,\ldots,100$	All

rameters for Stirmark used in this implementation is listed in Table 5. Various types of attacks in several levels have been applied. For example of AFFINE, 8 levels of affine transformed images are generated. Robustness is examined by extracting ω from transformed images.

Evaluation results of is shown below. Fingerprint is detected from 24 images out of 47 attacked images as shown in Table 5, labeled as "Succeed."

We also show how fingerprinted image in which *DCMP* affects robustness of *CCemb* by comparing *CCemb* with and without *DCMP*. The latter one shows $31/47$ cases are successfully extracted. The comparison of the two methods is shown in Fig. 13. Black lines show successful cases of the proposed one, *CCemb* with *DCMP* and gray lines show *CCemb* without *DCMP*. The experimental results provide effective evidence showing that the performance of *CCemb* is little affected by *DCMP*.

6. Conclusion

Pseudo-blind fingerprinting is presented in this paper with experimental results of practical and secure image trading system. We have accomplished our objective by proposing the alternative method that provides feasibility and robustness which are lacking in conventional blind methods. Feasibility and robustness have been satisfied by replacing encryption with one of the media process approaches, image decomposition. A proof of robustness is provided by evaluation result. Experimental results provide validity of our concept which is effectively used for a privacy-secure digital image trading system under which client's privacy can be protected at sufficient robustness and feasible processing cost. Our next task is to enhance decomposition and fingerprint embedding to develop the system so that it could be applied for practical application where a privacy-secure content trading is needed.

References

[1] *Shibboleth Project*. `http://shibboleth.internet2.edu`.

[2] Jun FURUKAWA. Secure detection of watermarks. *IEICE*, E87-A(1):212–220, Jan 2004.

[3] Hiroyuki INABA and Yukiko YAMAMOTO. Proposal on digital contents distribution system for protecting both privacy and copyrights. *IEICE*, J89-D(12):2536–2542, Dec 2006.

[4] MultiMedia Intelligence. Multimedia intelligence identifies digital watermarking and fingerprinting as key new opportunity, 2008. `http://www.multimediaintelligence.com/index.php?option=com_content&view=article&id=98:multimedia-intelligence-identifies-digital-wate37:frontpagetitleonly&Itemid=146`.

[5] Keiichi Iwamura, Kouichi Sakurai, and Hideki Imai. Blind fingerprinting. *Technical report of IEICE. ISEC*, 97:63–74, 1997.

[6] Keiichi Iwamura, Kouichi Sakurai, and Hideki Imai. A secure digital watermark system for secondary distribution. *IEICE. A*, 84(5):624–632, May 2001.

[7] Minoru Kuribayashi and Hatsukazu Tanaka. Fingerprinting protocol for images based on additive homomorphic prop-

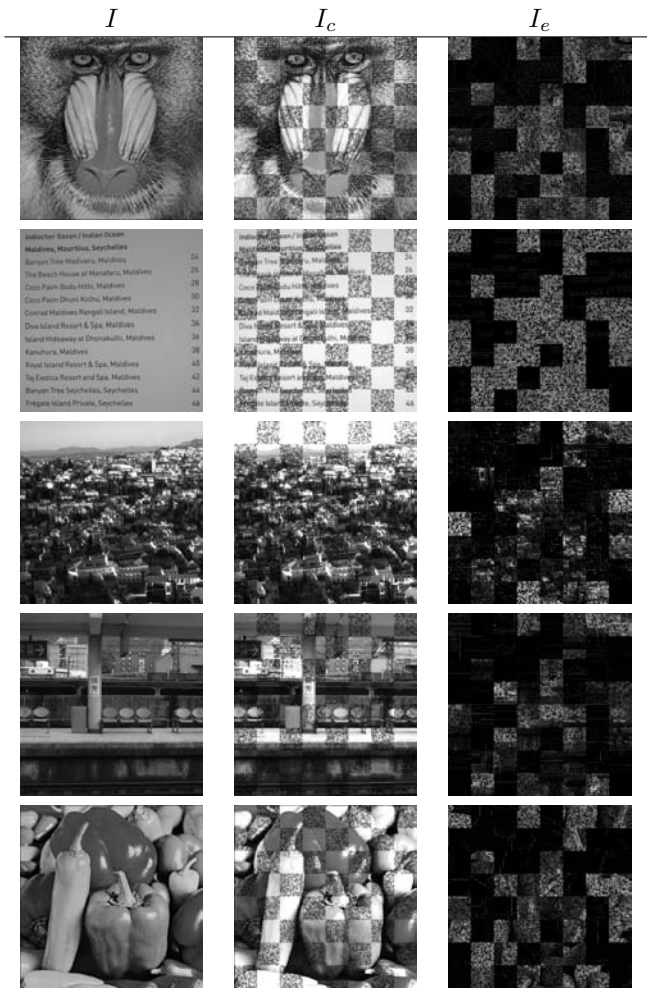


Fig. 10 Various Types of Output Images

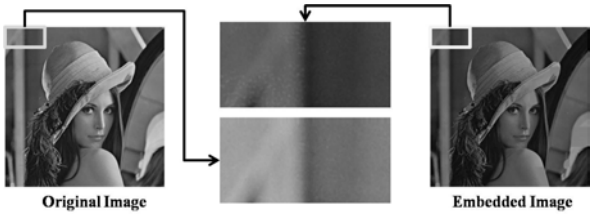


Fig. 11 Original Image I and Fingerprinted Image I'

erty. *IEEE Trans. Image Processing*, pages 2129–2139, 2005.

[8] Mitsuo Okada, Hiroaki Kikuchi, and Yasuo Okabe. Multi-bit embedding in asymmetric digital watermarking without exposing secret information. *IEICE*, E91-D(5):1348–1358, May 2008.

[9] Mitsuo OKADA, Yasuo OKABE, and Tetsutaro UEHARA. A privacy enhanced image sharing system on the sensing web based on a fingerprinting technique, Dec 2008.

[10] Mitsuo OKADA, Yasuo OKABE, and Tetsutaro UEHARA. A privacy-secure content trading system for small content providers using semi-blind digital watermarking. *The 2009 International Workshop on Forensics for Future Generation Communication environments (F2GC) in conjunction with CSA2009*, 0:0, Dec 2009.

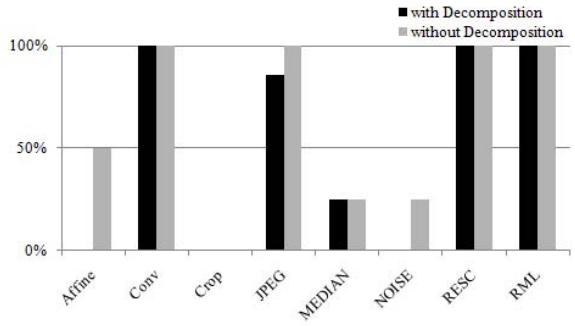


Fig. 13 Robustness of Watermark

[11] Mitsuo OKADA, Yasuo OKABE, and Tetsutaro UEHARA. Security analysis on privacy-secure image trading framework using blind watermarking, Jul 2009.

[12] Mitsuo OKADA, Yasuo OKABE, and Tetsutaro UEHARA. Semi-blind fingerprinting utilizing ordinary existing watermarking techniques. *The 8th International Workshop on Digital Watermarking (IWDW2009)*, 5703:14–28, Aug 2009.

[13] Mitsuo OKADA, Yasuo OKABE, and Tetsutaro UEHARA. A web-based privacy-secure content trading system for small content providers using semi-blind digital watermarking, Jan 2010.

[14] Fabien A. P. Petitcolas. Watermarking schemes evaluation. *IEEE Signal Processing*, 17(5):58–64, Sep 2000.

[15] Birgit Pfitzmann and Ahmad-Reza Sadeghi. Coin-based anonymous fingerprinting. *EUROCRYPT'99, LNCS 1592*, pages 150–164, 1999.

[16] Birgit Pfitzmann and Matthias Schunter. Asymmetric fingerprinting. *EUROCRYPT'96 LNCS*, 1070:84–95, Jan 1996.

[17] Birgit Pfitzmann and Michael Waidner. Anonymous fingerprinting. *EUROCRYPT'97 LNCS*, 1233:88–102, Jan 1997.

[18] J. P. Prins, Z. Erkin, and R.L. Lagendijk. Anonymous fingerprinting with robust qim watermarking techniques. *European Journal of Information Systems*, 2007:1–13, 2007.

[19] Fabien A.P. Petitcolas Stefan Katzenbeisser. *Information Hiding Techniques for Steganography and Digital Watermarking*. Artech House, 2000.